IN THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) A method for detecting and processing attacks on a computer network, comprising:

receiving a plurality of successive sets of data each set each set being associated with a corresponding security event associated with the computer network;

placing each set of data in a selected one of a plurality of queues based at least in part on a queue selection algorithm by which sets of data associated with related security events are grouped into the same queue while sets of data associated with unrelated security events are spread across different queues; and

processing the sets of data by:

(1) selecting for processing a first set of data from a first queue;

(2) selecting for processing a next set of data from a next queue in order that contains a set of data; and

(3) repeating step (2) until no queue contains a set of data that has not yet been selected for processing;

wherein the queue selection algorithm includes performing a first computation on a first at least a portion of the set of data to obtain a first index, performing a second computation on a second at least a portion of the set of data to obtain a second index, and selecting a queue for the set of data being processed based at least in part on the first and second indices, such that sets of data having the same values for both the first at least a portion of the set of data and the second at least a portion of the set of data are placed in the same queue;

whereby a critical set of data associated with a critical security event is timely selected for processing even under circumstances in which numerous sets of data associated with a corresponding set of security events that are related to each other but not to the critical security event are received prior to the critical set of data being received.

2. (Original) The method of claim 1, wherein the step of processing includes

determining the responsive action to be taken.

3. (Previously Presented) The method of claim 1, wherein the step of processing includes taking action in response to the data.

4. (Original) The method of claim 1, further comprising sending an alert concerning the data to a recipient in the administrative domain in which the data was received.

5. (Original) The method of claim 4, wherein the alert comprises an e-mail message to an individual.

6. (Original) The method of claim 4, wherein the alert comprises activating a pager.

7. (Original) The method of claim 1, wherein the data is received at a first system and further comprising sharing information concerning the data with a second system in the same administrative domain as the first system.

8. (Previously Presented) The method of claim 1, further comprising sending via a trusted third party a handoff message comprising information concerning the data to an administrative domain other than the administrative domain in which the data was received.

9. (Original) The method of claim 8, wherein the handoff message is sent directly to the other administrative domain.

10. (Canceled)

11. (Original) The method of claim 8, wherein the handoff message is generated and sent automatically, without human intervention.

12. (Original) The method of claim 1, further comprising scanning data arriving on at least one port.

13. (Original) The method of claim 12, wherein the at least one port is a switch port and the scanning comprises copying the data passing the at least one port to a copy port associated with the switch.

14. (Original) The method of claim 13, wherein the scanning further comprises dynamically changing the port being scanned.

15. (Original) The method of claim 12, wherein the scanning comprises sending a network management protocol request.

16. (Original) The method of claim 12, wherein the scanning comprises searching the data passing the at least one port for a string.

17. (Original) The method of claim 12, wherein the scanning comprises searching the data passing the at least one port for a type of message.

18. (Previously Presented) The method of claim 12, wherein the scanning comprises searching the data passing the at least one port for an attempt to access a service identified as vulnerable to attack.

19. (Original) The method of claim 18, wherein the service is the telnet service.

20. (Original) The method of claim 1, further comprising classifying the data.

21. (Original) The method of claim 1, further comprising classifying the data by type of attack.

22. – 25. (Canceled)

26. (Previously Presented) The method of claim 1, wherein the plurality of queues is organized into a table of queues having at least one row and at least one column.

27. (Currently Amended) The method of claim 26, wherein the table has R rows and C columns and the selected queue is determined by calculating a row address equal to the modulus R of the[[a]] first index quantity associated with the data and a column address equal to the modulus C of the[[a]] second index quantity associated with the data.

28. (Currently Amended) The method of claim 1[[27]], wherein the first quantity index is a hash value of a message containing the data.

29. (Currently Amended) The method of claim 1[[27]], wherein the second ~~quantity~~ index is a hash value of the source address of a message containing the data.

30. (Canceled)

31. (Previously Presented) The method of claim 1, further comprising associating a security event with at least one other security event to which it is related.

32. (Previously Presented) The method of claim 1, wherein a first security event is associated with a second security event if the first security event and second security event are associated with the same message.

33. (Previously Presented) The method of claim 1, wherein a first security event is associated with a second security event if the first security event and second security event are associated with the same sub-network and the security event rate for that sub-network exceeds a statistically determined baseline security event rate for the sub-network.

34. (Original) The method of claim 1, wherein the processing includes identifying the source of an attack.

35. (Original) The method of claim 1, wherein the processing includes tracking messages associated with an attack back to identify a point of attack at which messages associated with the attack are entering the network.

36. (Original) The method of claim 35, wherein the tracking comprises:

providing a network topology map, the map including information concerning the devices that comprise the network and the ports associated with each respective device;

scanning the ports of a first device for messages associated with the attack, wherein the first device is the device from which the data associated with the attack was first received; and

identifying a first port in the first device as the port at which at least one message associated with the attack was received by the first device.

37. (Original) The method of claim 36, wherein the tracking further comprises:

determining whether the first port is an external connection to another administrative domain;

in the event that the first port is an external connection, identifying the first port as the point of attack; and

in the event that the first port is not an external connection:

identifying a second device to which the first device is connected via the first port;

scanning the ports of the second device for messages associated with the attack; and

identifying a second port in the second device as the port at which at least one message associated with the attack was received by the second device.

38. (Original) The method of claim 37, wherein successive iterations of the steps recited in claim 37 are repeated until a port is identified as the point of attack.

39 - 40 (Canceled)

41. (Currently Amended) A system for detecting and processing attacks on a computer network, comprising:

a processor configured to:

receive a plurality of successive sets of data each set each set being associated with a corresponding security event associated with the computer network;

place each set of data in a selected one of a plurality of queues based at least in part on a queue selection algorithm by which sets of data associated with related security

events are grouped into the same queue while sets of data associated with unrelated

security events are spread across different queues; and

process the sets of data by:

(1) selecting for processing a first set of data from a first queue;

(2) selecting for processing a next set of data from a next queue in order

that contains a set of data; and

(3) repeating step (2) until no queue contains a set of data that has not yet

been selected for processing; and

a memory associated with the processor and configured to store said plurality of

successive sets of data and in said plurality of queues;

wherein the queue selection algorithm includes performing a first computation on a first

at least a portion of the set of data to obtain a first index, performing a second computation on a

second at least a portion of the set of data to obtain a second index, and selecting a queue for the

set of data being processed based at least in part on the first and second indices, such that sets of

data having the same values for both the first at least a portion of the set of data and the second at

least a portion of the set of data are placed in the same queue;

whereby a critical set of data associated with a critical security event is timely selected

for processing even under circumstances in which numerous sets of data associated with a

corresponding set of security events that are related to each other but not to the critical security

event are received prior to the critical set of data being received.

42. (Currently Amended) A system for detecting and processing attacks on a computer

network, comprising:

means for receiving a plurality of successive sets of data each set each set being

associated with a corresponding security event associated with the computer network;

means for placing each set of data in a selected one of a plurality of queues based at least

in part on a queue selection algorithm by which sets of data associated with related security

events are grouped into the same queue while sets of data associated with unrelated security events are spread across different queues; and

means for processing the sets of data by:

(1) selecting for processing a first set of data from a first queue;

(2) selecting for processing a next set of data from a next queue in order that contains a set of data; and

(3) repeating step (2) until no queue contains a set of data that has not yet been selected for processing;

wherein the queue selection algorithm includes performing a first computation on a first at least a portion of the set of data to obtain a first index, performing a second computation on a second at least a portion of the set of data to obtain a second index, and selecting a queue for the set of data being processed based at least in part on the first and second indices, such that sets of data having the same values for both the first at least a portion of the set of data and the second at least a portion of the set of data are placed in the same queue;

whereby a critical set of data associated with a critical security event is timely selected for processing even under circumstances in which numerous sets of data associated with a corresponding set of security events that are related to each other but not to the critical security event are received prior to the critical set of data being received.

43. (Currently Amended) A computer program product for detecting and processing attacks on a computer network, the computer program product being embodied in a computer readable medium and comprising computer instructions for:

receiving a plurality of successive sets of data each set each set being associated with a corresponding security event associated with the computer network;

placing each set of data in a selected one of a plurality of queues based at least in part on a queue selection algorithm by which sets of data associated with related security events are

grouped into the same queue while sets of data associated with unrelated security events are spread across different queues; and

processing the sets of data by:

(1) selecting for processing a first set of data from a first queue;

(2) selecting for processing a next set of data from a next queue in order that contains a set of data; and

(3) repeating step (2) until no queue contains a set of data that has not yet been selected for processing;

wherein the queue selection algorithm includes performing a first computation on a first at least a portion of the set of data to obtain a first index, performing a second computation on a second at least a portion of the set of data to obtain a second index, and selecting a queue for the set of data being processed based at least in part on the first and second indices, such that sets of data having the same values for both the first at least a portion of the set of data and the second at least a portion of the set of data are placed in the same queue;

whereby a critical set of data associated with a critical security event is timely selected for processing even under circumstances in which numerous sets of data associated with a corresponding set of security events that are related to each other but not to the critical security event are received prior to the critical set of data being received.